CUSTOMER NUMBER **25268**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants:	Chris Yoochang Chung et al.	Attorney Docket No:	UNIV0123
Serial No:	10/053,431	Group Art Unit:	2676
Filed:	January 17, 2002	Examiner:	Nguyen, Hau H.
Title:	PROGRAMMABLE 3D GRAPHICS PIPELINE FOR MULTIMEDIA APPLICATIONS		

PETITION FOR WITHDRAWAL OF RESTRICTION REQUIREMENT UNDER 37 CFR 1.144

Bellevue, Washington 98004

December 6, 2004

TO THE DIRECTOR OF THE PATENT AND TRADEMARK OFFICE:

REMARKS

Status of Restriction Requirement

In the final Office Action dated September 3, 2004, the Examiner responded to applicants' election and traverse of the Restriction requirement, making the Restriction final. Applicants hereby petition the Director to withdraw the Restriction requirement for the reasons set forth below. The petition fee is enclosed herewith. A listing of the claims as currently amended in the Response to the Final Office Action that is being filed concurrently herewith is presented below for the convenience of the Director.

The Examiner has restricted the application into four groups as follows:

Group I, Claims 1, and 3-16, which the Examiner asserts are "drawn to 'programmable graphics pipeline' classified in class 345, subclass 506;"

Group II, Claims 17-22, which the Examiner asserts are "drawn to 'processing graphics data and media data' classified in class 345, subclass 546;"

Group III, Claims 23-29, which the Examiner asserts are "drawn to 'processing variable length data' classified in class 345, subclass 522;"

Group IV, Claim 30, which the Examiner asserts is "drawn to 'caching texture data and media data' classified in class 345, subclass 552.

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

Amendment to the Claims

1. (Currently Amended) A programmable graphics pipeline for processing multimedia data comprising:

(a) an instruction cache for storing at least graphics and media instructions;

(b) a first register file for storing the multimedia data and intermediate data;

(c) a first vector functional unit in communication with the instruction cache and the register file, said vector functional unit performing at least graphics and media instructions, to produce at least graphics and media data; and

(d) an enhanced texture cache in communication with the first vector functional unit, the first vector functional unit obtaining from said enhanced texture cache a vector of at least one partition of the multimedia data, wherein said enhanced texture cache is not limited to storage of one type of multimedia data.

2. (Previously Cancelled)

3. (Previously Presented) The programmable graphics pipeline of Claim 1, wherein the enhanced texture cache comprises:

(a) a line buffer that provides multiple read ports for accessing the multimedia data; and

(b) a cache area in communication with the line buffer, said cache area storing the multimedia data.

4. (Original) The programmable graphics pipeline of Claim 3, further comprising an enhanced texture address unit in communication with the enhanced texture cache, said enhanced texture address unit being employed for converting multiple inverse-mapped source coordinates into absolute memory addresses of the multimedia data in an arbitrary-sized block.

5. (Original) The programmable graphics pipeline of Claim 4, wherein the enhanced texture address unit further generates filter coefficients for the multiple inverse-mapped source coordinates for one of a bilinear filtering and a trilinear filtering of the data.

///

///

///

///

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

6. (Original) The programmable graphics pipeline of Claim 4, further comprising an enhanced rasterization unit in communication with the enhanced texture address unit, said enhanced rasterization unit being employed for generating:

(a) a plurality of destination coordinates for at least one of:

(i) a plurality of primitives being rendered; and

(ii) a media processing output; and

(b) a plurality of source coordinates for at least one of:

(i) texture data associated with the plurality of destination coordinates for the plurality of primitives; and

(ii) media data associated with the plurality of destination coordinates for the media processing.

7. (Original) The programmable graphics pipeline of Claim 6, wherein the plurality of primitives include at least one of:

(a) dots;

(b) lines;

(c) triangles;

(d) rectangles; and

(e) polygons.

8. (Original) The programmable graphics pipeline of Claim 6, further comprising an enhanced Z-buffer unit in communication with the enhanced rasterization unit and the enhanced texture address unit, said enhanced Z-buffer unit being employed for loading source coordinates of a primitive being rendered to provide the enhanced texture address unit with variable access to the source coordinates if perspective address generation cannot be used.

9. (Original) The programmable graphics pipeline of Claim 8, wherein the enhanced Z-buffer unit is further employed for determining a depth of a new source pixel in relation to an old source pixel.

10. (Original) The programmable graphics pipeline of Claim 1, further comprising a blending unit that is in communication with the first vector functional unit, said blending unit being employed for:

///

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

(a) combining a graphics data output from the first vector functional unit with a color value to produce a blended value; and

(b) combining the blended value with a destination pixel value.

11. (Original) The programmable graphics pipeline of Claim 1, further comprising an output buffer in communication with the first vector functional unit, said output buffer being employed for concatenating one of successive image outputs and successive video outputs from the first vector functional unit, to reduce a number of subsequent write transactions to a memory.

12. (Previously Presented) The programmable graphics pipeline of Claim 1, further comprising an output buffer in communication with the first vector functional unit and the enhanced texture cache, said output buffer being employed for:

(a) concatenating successive media processing output from the first vector functional unit to produce concatenated data;

(b) providing the concatenated data as input to the enhanced texture cache; and

(c) converting a compressed format of destination coordinates into a plurality of destination addresses for media processing output data.

13. (Original) The programmable graphics pipeline of Claim 10, further comprising a write buffer in communication with the blending unit, said write buffer sending multiple write transactions to a memory, to reduce page misses.

14. (Original) The programmable graphics pipeline of Claim 11, further comprising a write buffer in communication with the output buffer, said write buffer sending multiple write transactions to a memory, to reduce page misses.

15. (Original) The programmable graphics pipeline of Claim 1, further comprising a configuration register in communication with the first vector functional unit, said configuration register providing the first vector functional unit with partitioned data access parameters and a location of a current instruction thereby instructing the vector functional unit to perform one of a graphics process and a media process.

16. (Original) The programmable graphics pipeline of Claim 1, further comprising:

(a) a second vector functional unit in communication with the instruction cache and performing the graphics and media instructions being performed by the first vector functional unit, to produce parallel graphics and media data; and

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

(b) a second register file in communication with the second vector functional unit, said second register file storing additional multimedia data and additional intermediate data.

17. (Previously Presented) A method for producing one of graphics pixel data and media output data, comprising the steps of:

(a) obtaining configuration data from a host processor, said configuration data comprising a partitioned data size and a location of an instruction;

(b) performing graphics rendering processing on graphics texture data with a programmable graphics rendering pipeline to produce graphics pixel data when the partitioned data size and the instruction correspond to graphics processing; and

(c) performing media processing on media source data with the programmable graphics rendering pipeline to produce media output data when the partitioned data size and the instruction correspond to media processing.

18. (Previously Presented) The method of Claim 17, wherein the step of performing graphics rendering processing comprises the steps of:

(a) producing a destination coordinate;

(b) producing a mapped texture coordinate corresponding to the destination coordinate;

(c) producing a plurality of surrounding memory addresses in which surrounding texture data are stored, said surrounding texture data corresponding to a plurality of surrounding texture coordinates surrounding the mapped texture coordinate;

(d) retrieving the surrounding texture data from the plurality of surrounding memory addresses, and storing the surrounding texture data retrieved in a cache; and

(e) performing interpolation on the surrounding texture data in the cache to determine a texture value at the mapped texture coordinate as the graphics pixel data for the destination coordinate.

19. (Original) The method of Claim 17, wherein the step of performing media processing corresponds to image processing comprises the steps of:

(a) producing a destination coordinate;

(b) producing at least one mapped image coordinate corresponding to the destination coordinate;

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

(c) producing at least one memory address where image data are stored corresponding to the at least one mapped image coordinate;

(d) retrieving the image data from the at least one memory address, and storing the image data retrieved in a cache; and

(e) performing an image manipulation function on the image data in the cache to produce image pixel data at the destination coordinate as the media output data.

20. (Original) The method of Claim 17, wherein the step of performing media processing corresponds to video processing comprises the steps of:

(a) producing a destination coordinate;

(b) producing at least one mapped video coordinate corresponding to the destination coordinate;

(c) producing at least one memory address where video data are stored corresponding to the at least one mapped image coordinate;

(d) retrieving the video data from the at least one memory address, and storing the video data retrieved in a cache; and

(e) performing a video manipulation function on the video data to produce video pixel data at the destination coordinate as the media output data.

21. (Previously Presented) The method of Claim 17, wherein the programmable graphics rendering pipeline is employed for executing at least one of a partitioned inner product instruction, a partitioned arithmetic instruction, a partitioned logic instruction, a data movement instruction, and a loop-control instruction, to produce at least one of the graphics pixel data and the media output data.

22. (Previously Presented) The method of Claim 17, further comprising the step of storing the one of the graphics pixel data and the media output data in memory.

23. (Previously Presented) A programmable graphics pipeline for multimedia applications that performs graphics and media functions, comprising:

(a) a vector streaming engine that accesses variable length data from a memory and writes pixel data to the memory, wherein the variable length data comprise one of graphic texture source data and media source data, and wherein the pixel data are one of graphics pixel data and media pixel data; and

///

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

(b) a vector processing engine in communication with the vector streaming engine, said vector processing engine generating the pixel data from the variable length data.

24. (Original) The programmable graphics pipeline of Claim 23, wherein the vector streaming engine comprises:

(a) a vector input unit in communication with the vector processing engine; and

(b) a vector output unit in communication with the vector processing engine.

25. (Previously Presented) The programmable graphics pipeline of Claim 24, wherein the vector input unit comprises:

(a) an enhanced rasterization unit that generates at least one of:

(i) a destination coordinate corresponding to one of a graphics pixel location and a media pixel location; and

(ii) a source coordinate for one of a graphics texture source, and a media source; and

(b) an enhanced texture address unit in communication with the enhanced rasterization unit, said enhanced texture address unit generating a memory address corresponding to the source coordinate, said memory address defining a storage location for one of the graphics texture source data and the media source data; and

(c) a texture cache in communication with the enhanced texture address unit, said texture cache storing said one of the graphics texture source data and the media source data for use by the vector processing engine.

26. (Original) The programmable graphics pipeline of Claim 24, wherein the vector output unit comprises:

(a) a blending unit in communication with the vector processing engine, said blending unit performing blending operations on graphics pixel data;

(b) an output buffer in communication with the vector processing engine, said output buffer being employed for concatenating a plurality of media pixel data; and

(c) a write buffer in communication with both the blending unit and the output buffer, said write buffer writing one of the graphics pixel data and the media pixel data to memory in burst groups.

///

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

27. (Previously Presented) The programmable graphics pipeline of Claim 23, wherein the vector processing engine comprises:

(a) a first vector functional unit in communication with the vector streaming engine, said first vector functional unit being usable for graphics rendering processing functions and media processing functions;

(b) an instruction cache in communication with the vector functional unit, said instruction cache storing at least one machine instruction to be executed by the vector functional unit; and

(c) a first register file in communication with the first vector functional unit, said first register file storing intermediate data for use by the first vector functional unit.

28. (Original) The programmable graphics pipeline of Claim 27, wherein the vector processing engine comprises:

(a) a second vector functional unit in communication with the vector streaming engine, and in communication with the instruction cache, said second vector functional unit being usable for performing graphics processing functions and media processing functions in parallel with the first vector functional unit; and

(b) a second register file in communication with the second vector functional unit, said second register file storing intermediate data for use by the second vector functional unit.

29. (Original) The programmable graphics pipeline of Claim 23, further comprising a configuration register in communication with both the vector streaming engine and the vector processing engine, said configuration register providing a partitioned data size and a location of an instruction indicating operation of the programmable pipeline for one of graphics processing and media processing.

30. (Previously Presented) A method for performing one of a graphics function and a media function on variable length data with a programmable graphics pipeline, comprising the steps of:

(a) using a vector streaming engine:

(i) generating a plurality of output pixel coordinates;

(ii) generating a plurality of memory addresses corresponding to the output pixel coordinates at which are stored one of graphics texture data and media source data, wherein the graphics texture data and the media source data are not of equal length; and

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

(iii) caching said one of the graphics texture data and the media source data in a cache;

(b) passing data retrieved from the cache to a vector processing engine; and

(c) performing one of the graphics function and the media function on the data retrieved from the cache using the vector processing engine.

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

Examiner's Justification for the Restriction

The Examiner's justification for this four-way restriction is that the inventions of the four groups identified by the Examiner are related as subcombinations disclosed as usable together in a single combination, and the subcombinations are distinct from each other because it can be shown that they are separately usable. The Examiner reaches this conclusion by asserting that "invention I has separate utility such as "programmable graphics pipeline;" "invention II has separate utility such as "processing graphics data and media data;" invention III has separate utility such as "processing variable length data;" and "invention IV has separate utility such as "caching texture data and media data." MPEP § 806.05(d) requires that the Examiner provide an indication of the separate utility following the phrase "such as." In his justification of the restriction, the Examiner has simply repeated the phrase he stated to indicate to what each group of claims is drawn, in regard to the different subclasses in which the invention of each group is classified. However, the classification of the claims into four different subclasses as asserted by the Examiner is not justified and therefore, the phrase associated with that incorrect classification cannot be used as a basis for showing that each invention of each group is separately usable.

All four groups are indicated by the Examiner as falling under Class 345, which is broadly defined in the Manual of Patent Classification as: "Computer Graphics Processing, Operator Interface Processing, And Selective Visual Display Systems." Applicants agree that the Class for all of the claims in the application is indeed Class 345, but disagrees with the characterization of each group by the Examiner as falling in different subclasses.

The Examiner asserts that Claims 1 and 3-16 are classified in Class 345, subclass 506. The manual of patent classification indicates that subclass 506 is indented under subclass 502, which is indented under subclass 501. Subclass 501 broadly covers "A computer graphic processing system:" and more specifically, "subject matter comprising apparatus or method for processing or manipulating data for presentation by a computer prior to use with or in a specific display system." Subclass 502 broadly covers "Plural graphics processors:" and more specifically "subject matter wherein more than one graphics processor is used." Subclass 506 covers "Pipeline processors:" and more specifically, "subject matter wherein the plural processors are operated sequentially." So the Examiner is asserting that the Claims in Group I correspond to a computer graphics processing system having a plurality of graphics processors and more specifically, a plurality of pipeline

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

processors that are operated sequentially. However, applicants' Claim 1 as currently amended in the response to the final Office Action is as follows:

> A programmable graphics pipeline for processing multimedia data comprising:
>
> (a) an instruction cache for storing at least graphics and media instructions;
>
> (b) a first register file for storing the multimedia data and intermediate data;
>
> (c) a first vector functional unit in communication with the instruction cache and the register file, said vector functional unit performing at least graphics and media instructions, to produce at least graphics and media data; and
>
> (d) an enhanced texture cache in communication with the first vector functional unit, the first vector functional unit obtaining from said enhanced texture cache a vector of at least one partition of the multimedia data wherein said enhanced texture cache is not limited to storage of one type of multimedia data.

Nothing in Claim 1 refers to a plurality of processors or more specifically, to a plurality of pipeline processors that are operated sequentially. Accordingly, it appears that Claim 1 is not properly classified in subclass 506.

The Examiner indicates that the claims in Group II are "drawn to 'processing graphics data and media data' classified in class 345, subclass 546." Subclass 546 is indented under subclass 545, which is indented under subclass 530. Subclass 530 is broadly defined as "Computer Graphics Display Memory System:" and more specifically as "Subject matter wherein a storage system or display memory organization and structure is used for storing image data which is being created and processed for presentation." Subclass 545 is broadly defined as "Frame buffer:" and more specifically defined as "subject matter wherein the graphics display memory stores the contents of a screen of display image." Finally, subclass 546 is broadly defined as "Multi-format frame buffer:" and more specifically as "Subject matter wherein the frame buffer memory stores both video and graphics data, such as, YUV for video and RGB for graphics." Accordingly, a claim properly classified in subclass 546 would have to be drawn to a frame buffer storage system that stores both video and graphics data. However, Claim 17 is as follows:

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

A method for producing one of graphics pixel data and media output data, comprising the steps of:

(a) obtaining configuration data from a host processor, said configuration data comprising a partitioned data size and a location of an instruction;

(b) performing graphics rendering processing on graphics texture data with a programmable graphics rendering pipeline to produce graphics pixel data when the partitioned data size and the instruction correspond to graphics processing; and

(c) performing media processing on media source data with the programmable graphics rendering pipeline to produce media output data when the partitioned data size and the instruction correspond to media processing.

There is no mention in Claim 17 of a frame buffer storage system that stores both video and graphics data. Claim 17 recites ***producing one*** of graphics pixel data and media output data, but does NOT recite any frame buffer that stores two different types of data. Clearly, Claim 17 is not properly classified in subclass 546.

The Examiner asserts that the claims in Group III are classified in subclass 522, but broadly pertains to "Graphic command processing:" and more specifically is directed to "subject matter wherein a CPU or a host computer issues a command to a graphic processing system to perform an operation." Claim 23 is as follows:

A programmable graphics pipeline for multimedia applications that performs graphics and media functions, comprising:

(a) a vector streaming engine that accesses variable length data from a memory and writes pixel data to the memory, wherein the variable length data comprise one of graphic texture source data and media source data, and wherein the pixel data are one of graphics pixel data and media pixel data; and

(b) a vector processing engine in communication with the vector streaming engine, said vector processing engine generating the pixel data from the variable length data.

Once again, Claim 23 is apparently improperly classified, since there is no mention of a CPU or host processor issuing commands to a graphic processing to perform an operation. Instead, the

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

graphics pipeline is programmable and carries out the functions recited based on the programs with which it is programmed. But there is no indication that a CPU or host processor supplies the commands to make the graphics processor carry out these functions.

Claim 30 (Group IV) is classified in subclass 552, which is broadly directed to "Texture memory:" and more specifically, to "subject matter wherein the graphics display memory is used for storing shading and other attribute information where the information is added to the 'surface' of a graphical image or object to mimic the surface detail of real objects." However, Claim 30 is as follows:

> A method for performing one of a graphics function and a media function on variable length data with a programmable graphics pipeline, comprising the steps of:
>
> (a) using a vector streaming engine:
>
> (i) generating a plurality of output pixel coordinates;
>
> (ii) generating a plurality of memory addresses corresponding to the output pixel coordinates at which are stored one of graphics texture data and media source data, wherein the graphics texture data and the media source data are not of equal length; and
>
> (iii) caching said one of the graphics texture data and the media source data in a cache;
>
> (b) passing data retrieved from the cache to a vector processing engine; and
>
> (c) performing one of the graphics function and the media function on the data retrieved from the cache using the vector processing engine.

It should be apparent that this claim is not specifically directed to storing shading and other attribute information, but is instead is directed to selectively performing one of two different types of functions with a programmable graphics pipeline. Claim 30 also thus appears to be mischaracterized as belonging in subclass 552.

It appears that if the Examiner were to determine the proper subclass for each of the independent claims in this application, the claims would be searched in at most, two subclasses. However, since those two groups of claims are generally closely related, very little effort would be required in searching for prior art related to all of the claims in the application, and it would be

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314

preferable to search both subclasses anyway, regardless of which group of claims were elected. Indeed, the Examiner has apparently already made such a search. Accordingly, since there has been inadequate justification for issuing this restriction and the reasoning for doing so is not supported, the Director is respectfully requested to require the Examiner to withdraw the restriction and examine all claims in the application.

Respectfully submitted,

Ron Anderson

Ronald M. Anderson
Registration No. 28,829

RMA/SKM:lrg

I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid addressed to: Commissioner for Patents, Alexandria, VA 22313-1450, on December 6, 2004.

Kathy Pancin

Date: December 6, 2004

LAW OFFICES OF RONALD M. ANDERSON
600 - 108th Avenue N.E., Suite 507
Bellevue, Washington 98004
Telephone: (425) 688-8816 Fax: (425) 646-6314